W1077 EF

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-175009

(43)Date of publication of application : 21.06.2002

(51)Int.CI. G09C 1/00
G06F 17/60
G06F 19/00

(21)Application number : 2000-377990

(22)Date of filing : 07.12.2000

(71)Applicant : HITACHI LTD

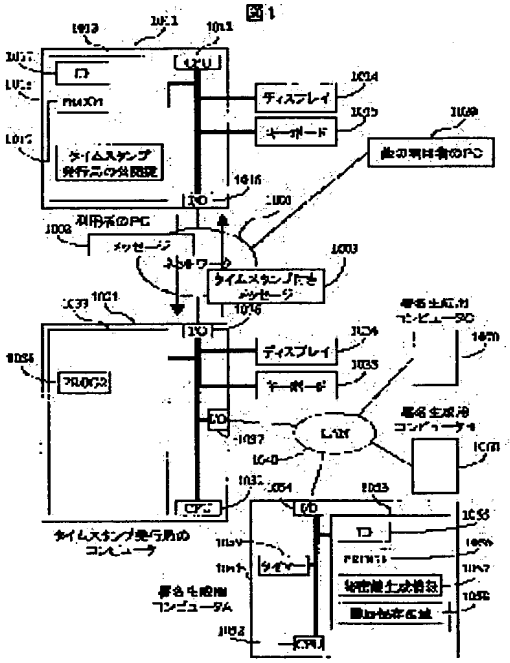
(72)Inventor : MIYAZAKI KUNIHIKO
SASAKI RYOICHI
TAKARAGI KAZUO
SUZAKI SEIICHI
MISHIMA HISANORI
MATSUKI TAKESHI
TAKEUCHI KUNIHITO
IWAMURA MITSURU
MATSUMOTO TSUTOMU

(54) METHOD FOR GENERATING DIGITAL SIGNATURE, AND METHOD FOR VERIFYING DIGITAL SIGNATURE

(57)Abstract:

PROBLEM TO BE SOLVED: To correctly and safely reflect the data related to a signature written in the past in a signing method, in which all two or more signature generating devices do not always necessarily generate the signature jointly.

SOLUTION: When signature is generated, the data to be used at generating of the signature next time are transmitted to other signature generating devices beforehand. Moreover, for generating the signature, at least one device is configured to be used continuously, so that the history data are shared at generation of the signature.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

W1077 EF

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報**（Ａ）　(11)特許出願公開番号
特開2002－175009
（P2002－175009A）
(43)公開日　平成14年６月21日(2002.6.21)

(51)Int.Cl.[7]	識別記号	ＦＩ	テーマコード（参考）
Ｇ０９Ｃ　1/00	６４０	Ｇ０９Ｃ　1/00	６４０Ｂ　５Ｊ１０４
			６４０Ｚ
	６２０		６２０Ａ
			６２０Ｚ
Ｇ０６Ｆ　17/60	ＺＥＣ	Ｇ０６Ｆ　17/60	ＺＥＣ

審査請求　未請求　請求項の数11　ＯＬ　（全 18 頁）　最終頁に続く

(21)出願番号　特願2000－377990(P2000－377990)

(22)出願日　平成12年12月７日(2000.12.7)

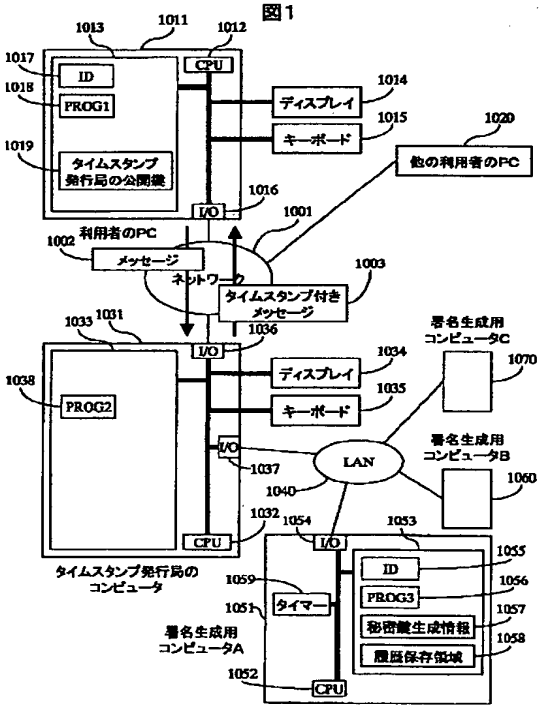
(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地
(72)発明者　宮崎　邦彦
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　佐々木　良一
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(74)代理人　100075096
弁理士　作田　康夫

最終頁に続く

(54)【発明の名称】　ディジタル署名生成方法およびディジタル署名検証方法

(57)【要約】
【課題】複数の署名生成装置のうちすべての装置が毎回連携して署名生成を行うとは限らない署名方法において、それ以前になされた署名に関わるデータを正しく、安全に反映させる。
【解決手段】署名生成に際し、次の署名生成の際に利用されるデータをあらかじめ他の署名生成装置に対し、送信する。また、署名生成に際し、少なくとも一つの装置は連続して利用されるように構成することにより、署名生成時に履歴データを共有する。

図1



【特許請求の範囲】
【請求項１】署名作成手段を備えたn台の装置を用いて、逐次、ディジタル署名を生成する方法であって、
j番目(j≧1)のディジタル署名生成時に、履歴データjを生成するステップと、
前記ｎ台の装置のうち、i番目(i>j)のディジタル署名生成に係わるm台(1≦m≦n)の装置において、
前記履歴データjを保持するステップと、
前記保持されたL個(1≦L<i)の履歴データj_1～j_Lのうち少なくとも一つを利用して、i番目のディジタル署名iを生成するステップと、を備えるディジタル署名生成方法。
【請求項２】請求項１において、
前記履歴データjは、前記j番目に生成されたディジタル署名jまたは、当該j番目に生成されたディジタル署名jの生成時に利用されたデータのいずれかであるディジタル署名生成方法。
【請求項３】請求項１において、
前記履歴データjを生成するステップは、前記m台の装置のいずれか自身において、実行されるものであるディジタル署名生成方法。
【請求項４】請求項１において、
前記m台の装置において、ディジタル署名を生成する際には、
前記m台の装置おのおのが、自身の持つ履歴データのうち最新のものを他のm-1台の装置に対し送信し、
他のm-1台の装置から送信された計m-1個の履歴データと、自身が保持する最新の履歴データをあわせた、合計m個の履歴データのなかから、最新の履歴データを選び、
当該最新の履歴データを、前記i番目のディジタル署名を生成する際に利用する履歴データのうちの一つとするディジタル署名生成方法。
【請求項５】請求項１において、
前記履歴データjを生成するステップは、前記n-m台の装置のいずれかにおいて、実行され、
前記保持するステップは、前記n-m台の装置のうちの少なくとも１台が前記履歴データjを前記m台の装置に送付するステップと、
前記m台の装置が前記送付された履歴データを保持するステップと、からなるディジタル署名生成方法。
【請求項６】請求項１において、
前記(i番目)新たなディジタル署名を生成するステップで利用される履歴データとして、履歴データ(i-1)と、少なくとも１つの履歴データk(k<i-1)を用いる、ディジタル署名生成方法。
【請求項７】請求項２ないし６いずれかにおいて生成されたディジタル署名を検証する方法であって、
ディジタル署名iの検証にあたっては、
前記i番目のディジタル署名を生成するステップにおいて利用された履歴データが、あらかじめ定められた規則を満たすように利用されていることを確認するステップを備える、ディジタル署名検証方法。
【請求項８】請求項２ないし６いずれかにおいて生成されたディジタル署名を検証する方法であって、
ディジタル署名iの検証にあたっては、
複数のディジタル署名h(h>i)の生成ステップにおいて生成された履歴データhが、当該ディジタル署名iの生成ステップにおいて生成された履歴データiを、あらかじめ定められた規則を満たすように利用していることを確認するステップを備えるディジタル署名検証方法。
【請求項９】請求項７または８において、
前記あらかじめ定められた規則とは、前記複数の履歴データのうちの少なくとも一つを利用すること、であるディジタル署名検証方法。
【請求項１０】請求項７または８において、
前記あらかじめ定められた規則とは、前記複数の履歴データのうちの、すべてを利用すること、であるディジタル署名検証方法。
【請求項１１】請求項７または８において、
前記規則が、定められるのは、システム稼動時、または、署名生成時、または、署名検証時、であるディジタル署名検証方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、マルチメディアデータの正当性を保証する技術に関する。
【０００２】
【従来の技術】電子的な文書などのディジタル化されたデータ（マルチメディアデータともいう）に、従来のサイン、印鑑に相当する機能を付与する技術であるディジタル署名が、電子商取引などにおけるネットワークの高度利用を可能にする技術として、注目されつつある。
【０００３】ディジタル署名技術に関する文献としては、たとえば以下のものがある。
【０００４】文献１：Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, "Handbook of Applied Cryptography" CRC Press, Inc. １９９７
文献２：Bruce Schneier, "Applied Cryptography Second Edition", JohnWiley & Sons, Inc. １９９６
文献３："Standerd Specifications for Public Key Cryptography (DraftVersion 11)" IEEE P1363, IEEE, July 1999
ディジタル署名技術では、ディジタル署名生成者は、署名対象となるディジタル化されたデータ（以下、メッセ

ージという）Mあるいはその特徴値（圧縮値または、メッセージダイジェストともいう）であるハッシュ値に、自身が秘密裏に保持する秘密鍵を作用させることで、メッセージMに対するディジタル署名Aを生成する。そして、メッセージMにディジタル署名Aを付して公開する。ディジタル署名検証者は、メッセージMに付されたディジタル署名Aを上記秘密鍵と対の公開鍵を作用させることで得た結果と、メッセージMあるいはそのハッシュ値とを比較する。両者が一致しない場合は、ディジタル署名Aが生成された後にメッセージMに何らかの改ざんが加えられた可能性がある。このため、両者が一致する場合に、ディジタル署名Aがメッセージに対してなされたものであることを認証する。

【０００５】また、メッセージがある時点で存在したことをディジタル署名を用いて保証する技術として、タイムスタンプ技術がある。この技術はメッセージとその時点における時刻情報を合わせたデータに対して、ディジタル署名を生成することにより、当該メッセージがその時点において存在したことを保証するものである。

【０００６】タイムスタンプ技術に関しては、上記文献２のP.75、"CHAPTER4 IntermediateProtocols, 4.1 TIMESTAMPING SERVICES"と

文献４：International Application Number PCT/US91/05386

文献５：International Application Number PCT/US99/19061

文献６：Ahto Buldas, Helger Lipmaa, and Berry Schoenmakers, "Optimally Efficient Accountable Time-Stamping"

文献７：Ahto Buldas, Peeter Laud, Helger Lipmaa, and Jan Villenmson, "Time-Stamping with Binary Linking Schemes"

に開示されている。

【０００７】また、故障等により装置の一部が利用不可となった時にも、安全に処理を継続することなどを目的として、複数のエンティティが共同して署名生成を行う場合に、そのうちの一定数のエンティティがそろえば正しい署名を生成可能であるが、それ未満では署名を正しく生成できない「しきい値署名」技術が提案されている。

【０００８】しきい値署名に関しては以下の文献に開示されている。

【０００９】文献８：M. Cerecedo, T. Matsumoto, and H. Imai, "Efficient and Secure Multiparty Generation of Digital Siganatures Based on Discrete Logarithms," IEICE Trans. Fundamentals, Vol. E76-A, No. 4, pp. 532-545, April 1993.

文献９：R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Robust Threshold DSS Signatures," In Proc. of Eurocrypt '96, LNCS 1070, Springer-Verlag, pp.354-371, 1996.

文献１０：C. Park and K. Kurosawa, "New ElGamal Type Threshold Digital Signature Scheme," IEICE Trans. Fundamentals, vol.E79-A, no.1, pp.86-93, Jan. 1996.

さらにメッセージにディジタル署名生成者自身が何らかの改ざんを加えて新たにディジタル署名を生成して元のメッセージおよびディジタル署名と置き換える様な不正な行為、を防止する技術が開発されている。

【００１０】該技術では、ディジタル署名生成者は、署名対象となるメッセージM_nあるいはそのハッシュ値と１つ前に生成したディジタル署名A_{n-1}の生成に関わるデータと時刻データを、自身が秘密裏に保持する秘密鍵を作用させることで、メッセージM_nに対するディジタル署名A_nを生成する。このようにすると、ディジタル署名A_nの次に生成されるディジタル署名A_{n+1}には、１つ前に生成したディジタル署名A_nの生成に関わるデータが反映される。このため、ディジタル署名生成者が自身が生成したメッセージM_nに何らかの改ざんを加えて新たにディジタル署名A_nを生成し、これらを元のメッセージM_nおよびディジタル署名A_nと置き換えるような不正な行為を行うと、ディジタル署名A_{n+1}との間で整合がとれなくなる。

【００１１】上述した、メッセージにディジタル署名生成者自身が何らかの改ざんを加えて新たにディジタル署名を生成して元のメッセージおよびディジタル署名と置き換える様な不正な行為、を防止する技術は、文献２、文献４にlinking Protocolとして開示されている。

【００１２】

【発明が解決しようとする課題】しかしながら、上記不正行為を防止する技術においては、複数の装置が共同して署名生成を行うことについては配慮されていなかった。

【００１３】より具体的には、上記不正行為を防止する技術を、上記しきい値署名技術を用いて実現しようとする場合、署名生成を行う複数の装置の中には、一つ前の署名生成に係わらなかった装置、すなわち、一つ前に生成された署名に関する情報を何ら持たない装置、が存在する可能性があるので、一つ前に生成された署名情報を反映できず、不正を防止できない可能性がある。

【００１４】また、上記従来技術に開示された方法においては、ディジタル署名A_{n+1}の生成時に用いられたディジタル署名A_nの生成に関わるデータが何らかの原因により失われた場合、不正を防止できない可能性がある。

【００１５】本発明は、複数の装置が共同してディジタル署名を生成する際においても、確実に不正を防止する技術を提供しようとするものである。

【００１６】また、以前に生成した署名を反映させるように行う方法であって、署名生成時に必ずしもすべての装置を必要としない方法を提供するものである。

【００１７】また、本発明は、ディジタル署名A_{n+1}の生成時に用いられたディジタル署名A_nの生成に関わるデータが何らかの原因により失われても、確実に不正を防止する技術を提供しようとするものである。

【００１８】すなわち、連鎖を形成している署名データの一部が失われても、当該失われたデータ以外の署名データ相互の前後関係を保証する手段を与える方法を提供しようとするものである。

【００１９】さらに、本発明は、上記方法を用いたサービスシステムや、そこで用いる装置、または、それらを機能的に実現するプログラムを提供しようとするものである。

【００２０】

【課題を解決するための手段】上記目的を達成するために、本発明は、複数の装置によって署名生成をする際に、連続で署名生成に係わる署名生成装置が少なくとも１台は存在するように署名生成を行う方法を提供する。本発明によれば、少なくとも一つの署名生成装置は、一つ前になされた署名生成に関わるデータを保持しているため、そのデータを利用した署名生成が可能となる。

【００２１】より具体的には、複数の装置によって生成された署名に関わるデータであって、次の署名生成に際し利用されるデータは、当該署名の生成に係わったか否かによらず、全ての署名生成装置が保持すればよい。また、安全な場所に保管された上記データを、すべての署名生成装置が安全な方法でアクセスすることにより共有してもよい。

【００２２】このように実施することにより、署名生成をどのような署名生成装置の組み合わせによって行っても、それらの署名生成装置は、一つ前の署名生成に関わるデータを保持していることになる。

【００２３】また、本発明は、署名生成の際に、それ以前に生成された署名に関わるデータを、複数利用し、各々の署名に関する連鎖（前後関係）を確認可能なように構成する。これにより、データの紛失や不正者の存在など何らかの原因によって、一つの連鎖を確認できなくなったとしても、他の連鎖を確認することで、過去から現在に至る署名の連鎖をとぎれにくくすることが可能になる。

【００２４】より具体的には、本発明は、署名作成手段を備えたn台の装置を用いて、逐次、ディジタル署名を生成する方法として、j番目($j \geq 1$)のディジタル署名生成時に、履歴データjを生成するステップと、上記ｎ台の装置のうち、i番目($i>j$)のディジタル署名生成に係わるm台($1 \leq m \leq n$)の装置において、上記履歴データjを保持するステップと、上記保持されたL個($1 \leq L < i$)の履歴データ$j_1 \sim j_L$のうち少なくとも一つを利用して、i番目のディジタル署名iを生成するステップと、を提供する。

【００２５】上記履歴データjは、上記j番目に生成されたディジタル署名jまたは、当該j番目に生成されたディジタル署名jの生成時に利用されたデータのいずれかでもよい。

【００２６】また、上記履歴データjを生成するステップは、上記m台の装置のいずれか自身において、実行されるものでもよい。

【００２７】また、上記i番目($i>j$)のディジタル署名生成に係わるm台($1 \leq m \leq n$)の装置において、ディジタル署名を生成する際には、上記m台の装置おのおのが、自身の持つ履歴データのうち最新のものを他のm-1台の装置に対し送信し、他のm-1台の装置から送信された計m-1個の履歴データと、自身が保持する履歴データの中で最新のもの１個をあわせた、合計m個の履歴データのなかから、最新の履歴データを選び、当該最新の履歴データを、上記i番目のディジタル署名を生成する際に利用する履歴データのうちの一つとしてもよい。

【００２８】また、上記履歴データjを生成するステップは、上記n-m台の装置のいずれかにおいて、実行され、上記保持するステップは、上記n-m台の装置のうちの少なくとも１台が上記履歴データjを上記m台の装置に送付するステップと、上記m台の装置が上記送付された履歴データを保持するステップとからなるものでもよい。

【００２９】また、上記(i番目)新たなディジタル署名を生成するステップで利用される履歴データとして、履歴データ(i-1)と、少なくとも１つの履歴データk($k<i-1$)を用いるものでもよい。

【００３０】本発明は、上記において生成されたディジタル署名iの検証にあたり、上記i番目のディジタル署名を生成するステップにおいて利用された履歴データが、あらかじめ定められた規則を満たすように利用されていることを確認するステップ、または、複数のディジタル署名h($h>i$)の生成ステップにおいて生成された履歴データhが、当該ディジタル署名iの生成ステップにおいて生成された履歴データiを、あらかじめ定められた規則を満たすように利用していることを確認するステップを提供する。

【００３１】上記あらかじめ定められた規則とは、上記複数の履歴データのうちの少なくとも一つを利用すること、または、上記複数の履歴データのうちのすべてを利用するものでもよい。

【００３２】さらに、上記規則が、定められるのは、システム稼動時、または、署名生成時、または、署名検証時であってもよい。

【００３３】
【発明の実施の形態】（第１の実施例）本発明をネットワークを介したタイムスタンプサービスに適用した一実施例を図を用いて説明する。
【００３４】図１は、本実施例におけるシステム概略構成図である。ネットワーク 1001を介して、タイムスタンプ発行局の制御用コンピュータ 1031 と、何人かの利用者のPC 1011, 1020 が接続されている。利用者は、作成したメッセージが、確かにその時刻に存在したことを、後になって証明することができるようにするために、利用者のPC 1011 により当該メッセージ 1002 をネットワーク 1001 を介して、タイムスタンプ発行局に送る。タイムスタンプ発行局のコンピュータ1031は、利用者からメッセージ 1002 を受け取ると、当該メッセージ 1002 に対する時刻保証情報であるタイムスタンプを発行し、メッセージ1002とタイムスタンプとからなるタイムスタンプ付きメッセージ1003を利用者のPC 1011に送り返す。利用者のPC 1011は、当該タイムスタンプ付きメッセージ1003を受け取る。
【００３５】本実施例においては、タイムスタンプとして上述のLinking Protocolに基づく技術を利用する。すなわち、タイムスタンプ発行局は、メッセージと時刻情報に加えて、直前に発行したタイムスタンプに関するデータを付加した情報に対し、ディジタル署名を生成する。
【００３６】さらに本発明においては、ディジタル署名を生成する際に、一層の安全性および信頼性向上を図るために、しきい値署名技術を利用する。より具体的には、n台のコンピュータ（署名生成装置）を使いそのうちの任意のk台がそろった時には正しく署名生成を行うことができるが、n-k台では正しく署名生成することができないk-out-of-nしきい値署名方式を利用する。しきい値署名については、上記文献８～文献１０に開示されているほか、
文献１１：特願平11-307993号
に、署名生成過程で秘密通信を必要としないことを特徴とするしきい値署名が開示されている。しきい値署名技術を利用することにより、例えばn台の署名生成装置のうちのn-k台が何らかの原因により故障した場合にも、残りのk台を使ってディジタル署名の生成が可能であり、また一方で、n台の署名生成装置のうちのn-k台が、万一何らかの原因により内部に保存された秘密に保持すべき情報が漏洩したとしても、その情報からだけでは、ディジタル署名の偽造などの不正を行うことは困難なので、安全性と信頼性の高いタイムスタンプ発行システムを構築することが可能となる。
【００３７】以下の実施例では、2-out-of-3しきい値署名技術を例示するが、要求される安全性、信頼性、価格対性能比などに応じて、k, nを選ぶことができる。
【００３８】利用者のPC 1011 は、CPU 1012、メモリ 1013、I/O 1016、ディスプレイ 1014、キーボード 1015 とこれらを接続するバスによって構成され、また、I/O 1016を介してネットワーク 1001に接続されている。メモリ 1013内には、利用者のID1017 、プログラム PROG1 1018 、および、タイムスタンプ発行局の公開鍵 1019 が保存されている。プログラムは、バスを介してCPU 1012に伝達され、機能として具現化される。
【００３９】一方、タイムスタンプ発行局側は、タイムスタンプ発行局の制御用コンピュータ 1031、および、署名生成用コンピュータA 1051、コンピュータＢ 1060、コンピュータＣ 1070がLAN 1040を介して接続されている。また、LAN 1040は、利用するしきい値署名技術によっては、周知の暗号技術や物理的手段により内部を流れる情報が秘匿されるように構成されている必要がある場合もある。
【００４０】タイムスタンプ発行局側の制御用コンピュータ 1031は、CPU 1032、メモリ 1033、ネットワーク接続用I/O 1036、LAN接続用I/O 1037、ディスプレイ 1034、キーボード1035とこれらを接続するバスによって構成され、ネットワーク接続用I/O 1036を介してネットワーク 1001に接続され、また、LAN接続用I/O 1037を介してLAN 1040に接続されている。またメモリ 1033内にはプログラム PROG2 1038が保存されている。
【００４１】タイムスタンプ発行局側の署名生成用コンピュータA 1051、Ｂ 1060、Ｃ 1070は、基本的に同構成で、CPU 1052、メモリ 1053、LAN接続用I/O 1054、タイマー1059とこれらを接続するバスによって構成されており、さらにLAN接続用I/O 1037を介してLAN 1040に接続されている。タイマー 1059はあらかじめ何らかの方法により他の署名生成用コンピュータとの間で同期が取られている。またメモリ 1053内にはID 1055*、プログラム PROG3 1056、秘密鍵生成情報 1057が保存されている他、１つ以上前に生成されたディジタル署名または当該１つ以上前に生成されたディジタル署名の生成時に利用されたデータ（以下履歴データという）を保存するための履歴保存領域 1058が用意されている。なお、ID 1055、秘密鍵生成情報1057として、コンピュータ毎に固有の値が設定されている。
【００４２】上記、秘密鍵生成情報1057とは、各署名生成用コンピュータが秘密裏に保持すべきデータであり、あらかじめ定められた条件を満たす署名生成用コンピュータがそろった時（本実施例では2-out-of-3しきい値署名を用いているので、３台中の任意の２台のコンピュータがそろった時）には、それらのコンピュータが保持する秘密鍵生成情報からタイムスタンプ発行局の秘密鍵情報が復元できるように、あらかじめ分散保持されているものとする。分散保持するための具体的な方法については、例えば、上記文献８～文献１１に示されている。
【００４３】図２は、本実施例における処理フローの概略を示した図である。なお、フロー中のステップ2001、

2002、2003、2010、2011は利用者のPC 1011上のメモリ1013中に保存されたプログラムPROG1 1018を、CPU1012が実行することにより実現される機能である。同様に、ステップ 2004、2005、2007、2009は、タイムスタンプ発行局の制御用コンピュータ1031上のPROG2 1038をCPU1032が実行することによって実現される機能であり、ステップ 2006、2008はタイムスタンプ発行局の署名生成用コンピュータＡ 1051上のPROG 1056をCPU1052が実行することによって実現される機能である。

【００４４】「タイムスタンプ発行処理フロー」

ステップ2001：はじめ

[利用者のPCによる処理]

ｽﾃｯﾌﾟ 2002：存在時刻を証明したい文書を作成する

ステップ2003：タイムスタンプ発行局に対し、タイムスタンプの発行を依頼する（当該文書mのハッシュ値H_mおよび利用者のＩＤ I_nをタイムスタンプ発行局に送付）

[タイムスタンプ発行局の制御用コンピュータ 1031による処理]

ステップ2004：３台の署名生成用コンピュータから、利用可能な署名生成用コンピュータを２台選択する

ステップ2005：ステップ2004で選択された署名生成用コンピュータに、利用者のPCから受けとった文書を送信する

[タイムスタンプ発行局の署名生成用コンピュータ（上記ステップ2004で選択されたコンピュータ）による処理]

ステップ2006：選択された２台のコンピュータが協調して、当該文書に「時刻データ」「履歴データ」等を付加し、しきい値署名方法にしたがって署名（タイムスタンプ）を生成し、制御用コンピュータに送信する。なお、利用するしきい値署名方法によっては、署名生成用コンピュータと制御用コンピュータとの間で何回か通信が発生することもある

[タイムスタンプ発行局の制御用コンピュータ 1031による処理]

ステップ2007：タイムスタンプ発行局の署名生成用コンピュータで生成された署名（タイムスタンプ）の生成に利用したデータを、当該署名の生成に係わったか否かを問わずすべての署名生成用コンピュータに対して、最新の「履歴データ」として送信する

[タイムスタンプ発行局の署名生成用コンピュータ（すべてのコンピュータ）による処理]

ステップ2008：各署名生成用コンピュータは、制御用コンピュータから受け取ったデータを用いて、各コンピュータ内のメモリ内の履歴保存領域を更新し、正常に更新された旨、制御用コンピュータに送信する

[タイムスタンプ発行局の制御用コンピュータ 1031による処理]

ステップ2009：文書と生成された署名（タイムスタンプ）とからなるタイムスタンプ付きメッセージ1003を、依頼を行った利用者のPCに送信する［利用者のPCによる処理］

ステップ2010：利用者はタイムスタンプ発行局からタイムスタンプ付き文書を受け取り、タイムスタンプが正当なものであることを、タイムスタンプ発行局の公開鍵を用いて確認した後、当該タイムスタンプ付き文書を受け入れる

ステップ2011：おわり

本実施例によれば、上記ステップ2007において、生成された署名（タイムスタンプ）を生成する際に利用したデータを、最新の「履歴データ」として、すべての署名生成用コンピュータに送信しているため、どの署名生成用コンピュータも、署名生成時には、その時点での最新の履歴データを保持していることになる。したがって、署名生成にどの２台の署名生成装置を利用しても、正しく行うことができる。なお、上記ステップ2007では、すべての署名生成用コンピュータに履歴データを送信している。しかし、当該署名の生成に係わったコンピュータは、署名生成時にこれらのデータを入手可能であるので改めて送らなくてもよい。すなわち当該署名の生成に係わらなかったコンピュータのみに送信するようにしてもよい。

【００４５】図３は、しきい値署名として上記文献１１の一実施例に示された方法を利用した場合の上記Linking Protocolに基づくタイムスタンプを発行する場合の「タイムスタンプ発行処理フロー」ステップ2006における処理詳細フローを示した図である。なお、ここでは上記ステップ2004で選ばれた署名生成用コンピュータは、コンピュータA 1051およびコンピュータＢ 1060であるとする。またコンピュータI(I=A or B or C)の保持する秘密鍵生成情報で、コンピュータJと共に署名生成をする場合に利用するデータをe_{IJ}と書く。また、ベースとなるディジタル署名技術として、楕円曲線暗号を利用している。楕円曲線暗号で利用される下記のパラメータ、

定義体GF(p)の位数p（160-bit程度の奇素数）、

楕円曲線Eの定義式：$y^2=x^3+ax+b \bmod p$に現れるパラメータa,b、

上記楕円曲線E上のベースポイントPと呼ばれる点の座標：(x_p, y_p)、

上記ベースポイントPが生成する楕円曲線E上の有理点からなる群の位数：q（160-bit程度の奇素数）など、

および、楕円曲線上の加算、２倍算、スカラー倍演算、素数を法とした加算、減算、乗算、除算、べき乗演算などの基本演算機能については、あらかじめ各コンピュータが有しているものとする。これらのパラメータ、基本演算については、上記文献３に詳しい。

【００４６】［ステップ2006処理詳細フロー］

ステップ3001：はじめ

[制御用コンピュータによる処理]

ステップ3002：コンピュータA 1051、Ｂ 1060に文書mの

ハッシュ値H_j、および利用者のＩＤ I_jを送信（ステップ2005に相当）
[署名生成用コンピュータA 1051による処理]
ステップ3003A：履歴保存領域内に保持されていた「履歴データ」$I_{j-1}, H_{j-1}, T_{j-1}, L_{j-1}$から新たに$L_j=H(I_{j-1}, H_{j-1}, T_{j-1}, L_{j-1})$を計算する
[署名生成用コンピュータＢ 1060による処理]
ステップ3003B：コンピュータA 1051と同様に L_jを計算する
[署名生成用コンピュータA 1051による処理]
ステップ3004A：制御用コンピュータから受け取った、ハッシュ値H_j、および利用者のＩＤ I_jに、コンピュータA 1051のタイマーから取得した時刻T_j、コンピュータA 1051が履歴データとして保持している、インデックスj、Link Data L_j、および一つ前のタイムスタンプに関するデータ（利用者ID I_{j-1}、ハッシュ値H_{j-1}、時刻T_{j-1}）を結合し、署名対象データMを計算し、そのハッシュ値h(M)を計算する
[署名生成用コンピュータＢ 1060による処理]
ステップ3004B：ステップ3003Aと同様にしてh(M)を計算する
[署名生成用コンピュータA 1051による処理]
ステップ3005A：$0<k_A<q$を満たす自然数である乱数k_Aを生成し、楕円曲線上のスカラー倍演算 $R_A=k_A P$を計算し、コンピュータＢ 1060へ送信
[署名生成用コンピュータＢ 1060による処理]
ｽﾃｯﾌﾟ 3005B：$0<k_B<q$を満たす自然数である乱数k_Bを生成し、楕円曲線上のスカラー倍演算 $R_B=k_B P$を計算し、コンピュータA 1051へ送信
[署名生成用コンピュータA 1051による処理]
ステップ3006A：楕円曲線上の加算演算$(x,y)=R_A+R_B$、および、$r=x \pmod q$、$s_A=e_{AB}x+k_A h(M) \pmod q$を計算し、$r, s_A$、および、Mを制御用コンピュータに送信
[署名生成用コンピュータＢ 1060による処理]
ステップ3006B：楕円曲線上の加算演算$(x,y)=R_B+R_A$、および、$r=x \pmod q$、$s_B=e_{BA}x+k_B h(M) \pmod q$を計算し、$r, s_B$、および、Mを制御用コンピュータに送信（rはステップ3006Aでコンピュータ A 1051によって計算された値と同じになる）
[制御用コンピュータによる処理]
ステップ3007：$s=s_A+s_B \pmod q$を計算し、(r,s)をタイムスタンプTS_jとする
ステップ3008：おわり
図４は、図３と同様に、しきい値署名として上記文献１１の一実施例に示された方法を利用した場合の上記Linking Protocolに基づくタイムスタンプを発行する場合の「タイムスタンプ発行処理フロー」ステップ2007における処理詳細フローを示した図である。
【００４７】[ステップ2007処理詳細フロー]
ステップ4001：はじめ
[制御用コンピュータによる処理]
ステップ4002：すべての署名生成用コンピュータに対し、I_j, H_j, T_jを新しい履歴データとして送信し更新を依頼（ステップ2007に相当）
[署名生成用コンピュータA 1051、Ｂ 1060、Ｃ 1070による処理（それぞれ独立に行う）]
ステップ4003：受信した（I_j, H_j, T_j, L_j）を履歴格納領域に保存
ステップ4004：インデックスを１増やす
[制御用コンピュータによる処理]
ステップ4005：TS_jをタイムスタンプとして、$M=(j, I_j, H_j, T_j, I_{j-1}, H_{j-1}, T_{j-1}, L_j)$と共に、依頼を行った利用者（$I_j$）に返信（ステップ2009に相当）
[利用者のPCによる処理]
ステップ4006：タイムスタンプ付き文書を受信（ステップ2010に相当）
ステップ4007：おわり
図５は、図４に示した処理フロー中のステップ4004の直後の各署名生成用コンピュータ内の履歴保存領域の様子を示した図である。履歴保存領域には、インデックスj+1 5001、利用者ID I_j 5002、メッセージのハッシュ値H_j 5003、時刻情報T_j 5004、Link Data L_j 5005が保存されている。
【００４８】さらに、上記Linking Protocolでは、上記の一連の処理が終わった後、一つ前の利用者（利用者IDがI_{j-1}の利用者）に対し、連鎖が正しくつぎへ反映されているということをたどれるようにするために、今回の利用者のID（I_j）を伝える。
【００４９】タイムスタンプ付き文書を受信した利用者は、しきい値署名技術を利用していない場合と同様にタイムスタンプを検証することができる。具体的には、まず、タイムスタンプ発行局の公開鍵を使って、ディジタル署名検証技術により、$(j, I_j, H_j, T_j, I_{j-1}, H_{j-1}, T_{j-1}, L_j)$に対するタイムスタンプ(ディジタル署名) TS_jの正当性を確認する。タイムスタンプ発行局の不正などがないようにより詳細に確認するために、必要に応じて一つ前の利用者(利用者ID I_{j-1})に問い合わせ、L_{j-1}を入手することにより、受信したデータに含まれるLink data L_jが$H(I_{j-1}, H_{j-1}, T_{j-1}, L_{j-1})$に等しいかどうかを確認することができる。同様に一つ後の利用者(利用者ID I_{j+1})に問い合わせることにより、受信したデータから計算される値$H(I_j, H_j, T_j, L_j)$が、一つ後の利用者が受信したデータに含まれるLink data L_{j+1}に一致しているかどうかを確認することもできる。さらには、同じ操作を逐次繰り返すことにより２つ以上前の利用者や、２つ以上後の利用者に問い合わせることにより、タイムスタンプの正当性をより詳細に確認してもよい。
【００５０】以上に示したとおり、本実施例によれば、直前の署名生成に関するデータを、次の署名生成の際に反映させながら行うような署名生成法に対し、しきい値

署名のような複数の署名生成装置が関わる署名生成法を適用することが可能となる。

【００５１】なお、本実施例では、n台の装置のうちのk台がそろった時に正しく署名生成を行うことができる、k-out-of-nしきい値署名法を用いて説明を行ったが、これにとらわれず、例えば、４台の署名生成装置A,B,C,Dのうち A,Bの２台からなる組、または、B,C,Dの３台からなる組のどちらか一組がそろえば署名生成可能になるような分散署名技術に対しても同様に適用可能である。このような署名技術の一例が上記文献１１に示されている。

【００５２】また、k-out-of-nしきい値署名法の中には、k台以上の署名生成装置に署名生成を依頼し、そのうちの少なくともk台が正しく動作すれば、全体として正しい署名を生成するような技術もある。そのような技術を利用する場合には、例えば、上記ステップ2004で制御用コンピュータにおいて２台の署名生成用コンピュータを選ぶのではなく、すべての署名生成用コンピュータに対して、署名生成処理を依頼してもよい。

【００５３】さらに、上記実施例においては、制御用コンピュータと署名生成用コンピュータを別のコンピュータ装置によって実現していたが、署名生成用コンピュータのうちの１台、または、複数台によって、制御用コンピュータの役割を実現するようにしてもよい。利用者のPCから送信されるメッセージは、あらかじめ利用者のPCにおいてハッシュ関数を用いて計算されたハッシュ値であってもよい。

【００５４】上記実施例においては、次の点が要求される。

【００５５】１．署名生成に直接係わらない署名生成用コンピュータも、履歴データ更新のために通信可能な状態にあること、

２．各署名生成用コンピュータに対し、直前の署名生成に関わる履歴データを外部から設定すること

上記Linking Protocolなどの直前の署名に関わるデータを反映させながら次の署名生成を行う技術の安全性を保つためには、反映させるデータが、確かに直前の署名生成に関わるデータである、ということが保証されなければならないが、上記２において、各署名生成用コンピュータは外部から設定された履歴データが、確かに直前の署名生成に関わるデータであるかどうか、確認する必要がある。

【００５６】これらの条件を緩和する他の実施例を以下に示す。

【００５７】（第２の実施例）本実施例におけるシステム概略構成図は図１と同様である。本実施例で用いるしきい値署名方法（あるいはより一般に分散署名生成方法）には、次のような性質を持つ方法を用いる。

【００５８】[性質]任意のjに対し、j回目の署名生成に係わる署名生成用コンピュータの集合をSS_j、j+1回目の署名生成に係わる署名生成用コンピュータの集合をSS_{j+1}とすると、SS_jとSS_{j+1}との共通部分は空ではない（すなわち、２回連続で署名生成に係わる署名生成装置が少なくとも１台は存在する）

例えば、k>n/2の時、k-out-of-nしきい値署名は、署名生成に係わるk台の装置をどのように選んでも、少なくとも2k-n台の装置は２回連続で署名生成に係わることになるので、この性質を満たす。従って、第１の実施例で使った2-out-of-3しきい値署名もこの性質を満たす。

【００５９】本実施例では、上記の性質を持つ署名方法を利用することにより、以下に示すとおり、第１の実施例の場合のように署名生成に係わらなかった署名生成装置に対して履歴データを送信する必要がなくなる。すなわち、上記２の点が解決され、図２に示した「タイムスタンプ発行処理フロー」のうち、ステップ2007、2008が不要となる。

【００６０】本実施例における、ステップ2006の処理フローの詳細は、図６に示すとおりである。

【００６１】[ステップ2006処理詳細フロー]

ステップ6001：はじめ

[制御用コンピュータによる処理]

ステップ6002：ステップ3002と同様

[署名生成用コンピュータA 1051による処理]

ステップ6003A：コンピュータA 1051が保持するインデックスj_AをコンピュータＢ 1060に送信する

[署名生成用コンピュータＢ 1060による処理]

ステップ6003B：コンピュータＢ 1060が保持するインデックスj_BをコンピュータA 1051に送信する

[署名生成用コンピュータA 1051による処理]

ステップ6004A：$j_A \geqq j_B$ならＡが保持する「履歴データ」を、ステップ6005Aで利用すると共に、コンピュータＢ 1060に送信する。そうでなければ$j_A := j_B$とし、ステップ6005Aではコンピュータ Ｂ 1060から送信される「履歴データ」を利用する

[署名生成用コンピュータＢ 1060による処理]

ステップ6004B：$j_B \geqq j_A$ならＢが保持する「履歴データ」を、ステップ6005Bで利用すると共に、コンピュータA 1051に送信する。そうでなければ$j_B := j_A$とし、ステップ6005Bでコンピュータ A 1051から送信される「履歴データ」を利用する

[署名生成用コンピュータA 1051による処理]

ステップ6005A：ステップ3003A～3006Aと同様

[署名生成用コンピュータＢ 1060による処理]

ステップ6005B：ステップ3003B～3006Bと同様

[署名生成用コンピュータA 1051による処理]

ステップ6006A：インデックスを１増やし、I_j, H_j, T_j, L_jと共に履歴保存領域に保存する

[署名生成用コンピュータＢ 1060による処理]

ステップ6006B：インデックスを１増やし、I_j, H_j, T_j, L_jと共に履歴保存領域に保存する

[制御用コンピュータによる処理]
ステップ6007：TS_jをタイムスタンプとして、($j, I_j, H_j, T_j, I_{j-1}, H_{j-1}, T_{j-1}, L_j$)と共に、依頼を行った利用者($I_j$)に返信（ステップ2009に相当）
ステップ6008：おわり
本実施例においても、第１の実施例と同様にしてタイムスタンプ付き文書の正当性を確認することができる。

【００６２】本実施例においては、上記、[性質]に述べた分散署名技術を利用しているため、署名生成時に最新の履歴を知る署名生成用コンピュータが存在することが保証されている。さらに、上記ステップ6004Aおよびステップ6004Bにおいて、インデックスを比較することにより、どの署名生成用コンピュータが最新の履歴を有しているか判定可能である。より一般にk台の署名生成用コンピュータが署名生成に係わる場合にも同様に、記録されているインデックスが最大のコンピュータを調べることにより、最新の履歴を知るコンピュータを判定可能である。従って、署名生成に係わった署名生成用コンピュータが、履歴保存領域内の履歴データを更新することにより、署名生成に係わらない署名生成用コンピュータの履歴データまでは更新することなく、最新の履歴データを引き継いだ形で署名生成を行うことができるようになる。

【００６３】このように、本実施例によれば、しきい値署名を利用して、上記Linking Protocolなどの直前の署名に関わるデータを反映させながら次の署名生成を行う技術を実現することが可能となるので、実施例１とは異なり、署名生成に直接係わらない署名生成用コンピュータが、履歴データ更新のために通信可能な状態にある必要はない。すなわち、一部の装置が故障により利用不可となっても署名生成が可能であるという、しきい値署名の一つの特徴を維持することが可能である。

【００６４】さらに、履歴データとして利用されるデータ直前の署名生成に関わるは自分自身が保持しているデータ、または、他の署名生成用コンピュータから送られてきたデータのどちらかであるため、すべての署名生成用コンピュータが信頼できるならば、例えば、署名生成用コンピュータ間の相互認証技術、および、署名生成用コンピュータ間の通信路上でのデータ改竄を防ぐ技術などを適用することにより、直前の署名生成に関わる履歴データを外部から設定しなければならないという前述の条件２を満たすことができる。

【００６５】（第３の実施例）本実施例におけるシステム概略構成図は図１と同様である。また、本実施例で用いるしきい値署名方法（あるいはより一般に分散署名生成方法）には、上記、第２の実施例に示したのと同じ性質を持つ方法を用いる。

【００６６】本実施例では、以下のような履歴データの引き継ぎ方法を採用する。具体的には、第２の実施例では、署名生成の際に、その時点での最新の履歴データを引き継いでいた（連鎖を形成した）のに対し、本実施例では、当該署名（タイムスタンプ）の発行に係わるすべて署名生成装置各々がもつ当該署名装置にとっての履歴データを、当該署名（タイムスタンプ）の発行に係わるすべて署名生成装置が引き継ぐようにし、また、これらそれぞれの連鎖を独立に検証可能なように署名生成を行う。

【００６７】例えば、2-out-of-3しきい値署名を利用していて、次のような署名生成装置の組み合わせで署名生成（タイムスタンプ発行）を行ったとする。

【００６８】[タイムスタンプ発行シーケンス例]

１回目	Ａ	Ｂ	
２回目	Ａ		Ｃ
３回目		Ｂ	Ｃ
４回目	Ａ		Ｃ
５回目	Ａ		Ｃ

署名生成用コンピュータＢ 1060と署名生成用コンピュータＣ 1060とによる３回目の署名生成時には、署名生成用コンピュータＢ 1060にとっての最新の履歴データは１回目の署名生成に関するデータになり、署名生成用コンピュータＣ 1060にとっての最新の履歴データは２回目の署名生成に関するデータになる。本実施例においてはこれら両方の履歴データを用いたデータを３回目の署名生成に利用する。

【００６９】両方の履歴データを用いる場合には、例えば結合（連接）により一つにまとめて用いる方法を使うことが可能である。履歴データにはさらに識別情報（例えば、何回前の署名生成に関する履歴データであるか、など）を含めておいても良い。

【００７０】同様に、署名生成用コンピュータA 1051と署名生成用コンピュータＣ 1060とによる４回目の署名生成では、署名生成用コンピュータA 1051での２回目と署名生成用コンピュータＣ 1060での３回目の署名生成に関するデータを利用し、５回目の署名生成では、４回目の署名生成に関するデータを２つ合わせたデータを利用する。

【００７１】この署名生成方法により、生成される署名は、常に最新の履歴データが反映される、という第２の実施例による特徴に加えさらに、生成される署名には必ず各署名生成用コンピュータ自身が保持するデータ（すなわち各署名生成用コンピュータにとって信頼できるデータ）が反映するという特徴を備える。

【００７２】さらに本実施例において形成される複数の連鎖は、それぞれ独立に検証できるため、たとえ、当該署名の生成に係わるの中に悪意のある署名生成用コンピュータが存在し、正しくないデータを最新の履歴データとして送信したとしても、少なくとも一つの正しい連鎖が確認できるため、当該悪意のある署名生成用コンピュータによる悪影響を限定的にとどめることが可能となる。

【００７３】このことを上記のタイムスタンプ発行シーケンス例にしたがって以下に説明する。

【００７４】図１１は上記タイムスタンプ発行シーケンス時の連鎖の様子を模式的に表した図である。連鎖の様子 11001に示した矢印は、署名間の依存関係を示している。図１１に示されているとおり、４回目の署名生成には、２回目と３回目の署名生成に関わる履歴データが反映している。もし、この上記タイムスタンプ発行シーケンス例において、４回目の署名生成時に、署名生成用コンピュータＣ 1060が、署名生成用コンピュータA 1051に対して、正しい最新履歴データを送らなかったとする。この場合には、３回目の履歴データとの連鎖は確認できないことになる。しかし、この場合でもなお、４回目の署名生成に２回目の署名生成に関わる履歴データが正しく反映されていることは、正当な署名生成用コンピュータA 1051によって保証されるので、４回目以降の署名履歴と、２回目以前の署名履歴との連鎖は確認可能である。つまり署名生成用コンピュータＣ 1060が４回目の署名生成の際に行った不正行為による影響のおよぶ範囲は、３回目の署名と４回目の署名の不整合に限定される。

【００７５】従来技術である上記Linking Protocolなどでは、一個所（例えば３回目の署名と４回目の署名との間の）連鎖が切れてしまうと、３回目以前のいずれかの署名と、４回目以降のいずれかの署名との間には、何ら関係を見出すことができない。これはすなわち全体の信頼性を失うことにつながる可能性がある。これに対し、本実施例によれば上に述べた通りこの問題を解決できる。

【００７６】以上の通り、本実施例によれば、すべての署名生成用コンピュータが必ずしも信頼できるとは限らないような場面においても、更に効果的に上記２の条件を緩和することが可能となる。したがって、すべての署名生成用コンピュータが信頼できるとは限らなくても、その中の多くの署名生成用コンピュータが信頼できる場合には、本実施例に従うことにより、システム全体として信頼性の高いタイムスタンプシステムを構築することが可能となる。

【００７７】図７は、本実施例におけるステップ2006の処理フローの詳細を示したものである。

【００７８】[ステップ2006処理詳細フロー]

ステップ7001：はじめ

[制御用コンピュータ 1031による処理]

ステップ7002：ステップ3002と同様

[署名生成用コンピュータA 1051による処理]

ステップ7003A：インデックスj_AおよびＡの「履歴データ」をＢに送る

[署名生成用コンピュータＢ 1060による処理]

ステップ7003B：インデックスj_BおよびＢの「履歴データ」をＡに送る

[署名生成用コンピュータA 1051による処理]

ステップ7004A：（j_Aとj_Bのうち大きいほうをj、小さいほうをj_Sとする。）

Link data $L_j=H(I_{j\_S-1}, H_{j\_S-1}, T_{j\_S-1}, L_{j\_S-1}) || H(I_{j\_S-1}, H_{j\_S-1}, T_{j\_S-1}, L_{j\_S-1})$ を計算する（||は結合（コンカチネーション）をあらわす）

[署名生成用コンピュータＢ 1060による処理]

ステップ7004B：コンピュータA 1051と同様に、L_jを計算する

[署名生成用コンピュータA 1051による処理]

ステップ7005A：インデックスj、利用者のＩＤ I_j、ハッシュ値H_j、コンピュータA 1051のタイマーから取得した時刻T_j、および、j_S, I_{j_S-1}, H_{j_S-1}, T_{j_S-1}, j, I_{j-1}, H_{j-1}, T_{j-1}, L_jを結合し、署名対象データMを計算し、そのハッシュ値h(M)を計算する

[署名生成用コンピュータＢ 1060による処理]

ステップ7005B：コンピュータA 1051と同様に、h(M)を計算する

[署名生成用コンピュータA 1051による処理]

ステップ7006A：3005A～3006Aと同様

[署名生成用コンピュータＢ 1060による処理]

ステップ7006B：3005B～3006Bと同様

[署名生成用コンピュータA 1051による処理]

ステップ7007A：インデックスjを1増やし、I_j, H_j, T_j, L_jと共に履歴保存領域に保存

[署名生成用コンピュータＢ 1060による処理]

ステップ7007B：インデックスjを1増やし、I_j, H_j, T_j, L_jと共に履歴保存領域に保存

[制御用コンピュータによる処理]

ステップ7008：ステップ3008と同様

ステップ7009：TS_jをタイムスタンプとして、(j, I_j, H_j, T_j, j_S, I_{j_S-1}, H_{j_S-1}, T_{j_S-1}, j, I_{j-1}, H_{j-1}, T_{j-1}, L_j)と共に、依頼を行った利用者(I_j)に返信（ステップ2009に相当）

ステップ7010：おわり

さらに、上記Linking Protocolでは、連鎖が正しく次へ反映されていることを確認するために、署名生成用コンピュータAを直前に利用した利用者(利用者ID j_A)、および、署名生成用コンピュータB 1060を直前に利用した利用者(利用者IDj_B)に、今回の利用者ID(I_j)を送信する。

【００７９】本実施例に従い作成されたタイムスタンプ付き文書の正当性を確認するには、まず、タイムスタンプ発行局の公開鍵を使って、ディジタル署名検証技術により、(j, I_j, H_j, T_j, j_S, I_{j_S-1}, H_{j_S-1}, T_{j_S-1}, j, I_{j-1}, H_{j-1}, T_{j-1}, L_j)に対するタイムスタンプ(ディジタル署名) TS_jの正当性を確認する。正当性が確認されなければタイムスタンプTS_jは正当なものとは認められない。

【００８０】次に、タイムスタンプ発行局の不正などがないことをより詳細に確認するために、必要に応じて署

名生成用コンピュータAを直前に利用した利用者(利用者ID j_A)(注：j_A = j_S または jである)に問い合わせ、L_{jA-1}を入手することにより、受信したデータに含まれるLink data $L_j = H(I_{j_S-1}, H_{j_S-1}, T_{j_S-1}, L_{j_S-1}) || H(I_{j_S-1}, H_{j_S-1}, T_{j_S-1}, L_{j_S-1})$の前半分または後半分が$H(I_{jA-1}, H_{jA-1}, T_{jA-1}, L_{jA-1})$に等しいことを確認することができる。

【００８１】等しいことを確認できない場合は、署名生成用コンピュータAが故障または不正な動作を行っていた(外部から不正なデータを与えられたことにより、結果として不正な動作を行わさせられた場合を含む)、または署名生成用コンピュータAが保持していたデータ$I_{jA-1}, H_{jA-1}, T_{jA-1}, L_{jA-1}$が正しくない(あるいは正しく利用されない)などが原因である可能性があるので、もう一つの署名生成用コンピュータB 1060を用いた確認を行う。すなわち、署名生成用コンピュータB 1060を直前に利用した利用者(利用者ID j_B)に問い合わせ、L_{jB-1}を入手することにより、受信したデータに含まれるLink data L_jの前半分または後半分が$H(I_{jB-1}, H_{jB-1}, T_{jB-1}, L_{jB-1})$に等しいことの確認を行う。

【００８２】この作業で、等しいことが確認されれば、本検証方法において、タイムスタンプTS_jは正当なものと認められる(または、必要に応じて下記の更なる検証手段に進む)。そうでなければTS_jは正当なものとは認められない。

【００８３】署名生成用コンピュータ B1060を直前に利用した利用者(利用者ID j_B)の協力による確認は、署名生成用コンピュータAの影響を受けずに行うことができるため、すべての署名生成用コンピュータが必ずしも信頼できるとは限らないような場面、または、署名生成用コンピュータAの協力を得ることができない場合またはコンピュータAが不正な動作をしているときなどにおいても、更に効果的に上記２の条件を緩和することが可能となっていることがわかる。

【００８４】また、この検証方法によれば、検証時に必ずしも他の利用者が協力する（あるいは協力できる）とは限らないような場面においても検証を行いやすい、という利点もある。すなわち、もし署名生成用コンピュータAを直前に利用した利用者(利用者ID j_A)が問い合わせに応答せず、L_{jA-1}を入手することができなかった場合にも、署名生成用コンピュータB 1060を直前に利用した利用者(利用者ID j_B)の協力が得られれば、検証を行うことができる。

【００８５】さらに、署名生成用コンピュータAを次に利用した利用者に問い合わせることにより、受信したデータから計算される値$H(I_j, H_j, T_j, L_j)$が、署名生成用コンピュータAの一つ後の利用者が受信したデータに含まれるLink dataの前半分あるいは後半分に一致しているかどうかを確認することもできる。

【００８６】署名生成用コンピュータAが正しく動作していない、あるいは、署名生成用コンピュータAの一つ後の利用者の協力が得られないなどの原因により、Link dataとの一致が確認できない場合には、署名生成用コンピュータB1060の一つ後の利用者に問い合わせることにより、確認を行うこともできる。

【００８７】本検証方法によれば、上記方法の少なくともどちらか一方により後の利用者のLink dataとの一致を確認できれば正当なタイムスタンプとして認められる(または、必要に応じて下記の更なる検証手段に進む)。そうでなければ正当なタイムスタンプとは認められない。

【００８８】本実施例においても前記第１、第２の実施例と同様に、以上の手順を逐次繰り返すことにより２つ以上前の利用者や、２つ以上後の利用者に問い合わせることにより、タイムスタンプの正当性をより詳細に確認してもよい。

【００８９】（第４の実施例）図８は本発明を、
文献１２：特願2000－313123号
に開示された署名システムに適用した場合のシステム概略構成図である。

【００９０】文献１２では、正しく生成された署名であるか、不正に偽造にされた署名であるかを判定可能とするために、生成した署名の履歴を保管しておく技術が示されており、さらに、署名の生成にあたっては、一つ前に生成した署名そのものを次の署名生成時に反映させることにより、署名履歴に連鎖を形成させ、履歴を安全に保管することを容易にする技術が示されている。複数の装置（本実施例ではＩＣカード）を利用して、正しく連鎖が形成されるように署名生成を行うためには、基本的には、上記、第１～３の実施例と同様にして行うことができるが、本実施例では、上記第３の実施例と同様に行った場合を例示する。

【００９１】ネットワーク 8001を介して、署名者のコンピュータ 8031 と、何人かの購入者のPC 8011, 8020 が接続されている。購入者は、署名者からのメッセージ（例えば、電子的な画像、映像、音楽などのディジタル化されたマルチメディアデータや、電子的な（ディジタル化された）有価証券、契約書など）の購入（有償、無償を問わない）にあたり、署名者の署名付きメッセージ 8003 をネットワーク8001 を介してを受け取る。

【００９２】購入者のPC 8011 は、CPU 8012、メモリ 8013、I/O 8016、ディスプレイ 8014、キーボード 8015 によって構成され、これらがバスによって接続されており、また、I/O 8016を介してネットワーク 8001に接続されている。メモリ 8013内には、購入者のID 8017 、プログラム PROGI 8018 、および、署名者の公開鍵 8019が保存されている。プログラムは、バスを介してCPU 8012に伝達され、機能として具現化される。

【００９３】一方、署名者側は、署名者のコンピュータ 8031、および、署名者のＩＣカードＡ 8051、ＩＣカー

ドＢ 8060、ＩＣカードＣ 8070からなる。なお、本実施例でも、上記実施例と同じく2-out-of-3しきい値署名方法を利用する。そのため、ＩＣカードは３枚としているが、枚数は利用する署名スキームに応じて設定すればよい。その他の分散型署名技術を利用して実現することも可能である。

【００９４】署名者のコンピュータ 8031は、CPU 8032、メモリ 8033、I/O 8036、ディスプレイ 8034、キーボード 8035、外部記憶装置 8037、ＩＣカードリーダライタ 8038によって構成され、これらがバスによって接続されており、I/O 8036を介してネットワーク 8001に接続されており、ＩＣカードリーダライタ 8038を介して、ＩＣカードと通信できるようになっている。またメモリ 8033内にはプログラムPROG2 8038が保存されている。

【００９５】署名者のＩＣカードＡ 8051、Ｂ 8060、Ｃ 8070は、基本的に同構成で、CPU 8052、メモリ 8053、I/O 8054がバスによって接続されており、さらにI/O 8037を介して署名者のコンピュータと接続可能になっている。またメモリ 8053内にはID 8055、プログラム PROG3 8056、秘密鍵生成情報 8057が保存されている他、以前に生成したディジタル署名（タイムスタンプ）に関係するデータ（履歴データ）を保存するための履歴保存領域 8058が用意されている。なお、ID 8055、秘密鍵生成情報 8057は、コンピュータ毎に固有の値が保持されている。

【００９６】図９は本実施例における、概略処理フローである。

【００９７】「署名生成フロー」

ステップ9001：はじめ

[署名者のコンピュータによる処理]

ステップ9002：購入対象となるメッセージを作成

ステップ9003：３枚の署名生成用ＩＣカードから、利用可能な署名生成用ＩＣカードを２枚選択する

ステップ9004：ステップ9003で選択されたＩＣカードに、署名者のコンピュータから受けとった文書を送信する

[署名生成用ＩＣカード（上記ステップ9003で選択されたＩＣカード）による処理]

ステップ9005：選択された２枚のＩＣカードが協調して、当該文書に「履歴データ」等を付加し、しきい値署名方法にしたがって署名を生成し、署名者のコンピュータに送信する。なお、利用するしきい値署名方法によっては、署名者のコンピュータと署名用ＩＣカードとの間で何回か通信が発生することもある

[署名者のコンピュータによる処理]

ステップ9006：生成された署名を文書と共に、当該メッセージを購入する購入者に送信する

[購入者のPCによる処理]

ステップ9007：購入者は署名者から署名付き文書を受け取り、署名が正当なものであることを、署名者の公開鍵を用いて確認した後、当該署名付き文書を受け入れる

ステップ9008：おわり

図１０は、上記署名生成フロー中のステップ9005の詳細処理フローである。なお、上記ステップ9003で選択されたＩＣカードはＡ 8051とＢ 8060であったとする。

【００９８】ステップ10001：はじめ

[署名者のコンピュータ8031による処理]

ステップ10002：ステップ9004と同様

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10003A：ＩＣカードＡ 8051内に保持されたインデックスj_AおよびＡ 8051の「履歴データ」をＢ 8060に送る

[署名生成用ＩＣカードＢ 8060による処理]

ステップ10003B：ＩＣカードＢ 8060内に保持されたインデックスj_BおよびＢ 8060の「履歴データ」をＡ 8051に送る

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10004A：（j_Aとj_Bのうち大きいほうをj、小さいほうをj_Sとする。）

Link data $L_j = H(I_{j_S-1}, H_{j_S-1}, S_{j_S-1}, L_{j_S-1}) \| \| H(I_{j-1}, H_{j-1}, S_{j-1}, L_{j-1})$を計算する（||は結合（コンカチネーション）をあらわす）

[署名生成用ＩＣカードＢ 8060による処理]

ステップ10004B：ＩＣカードＡ 8051と同様に、Link data L_jを計算する

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10005A：j, Ij, Hj, Lj および j_S を結合し、署名対象データMとし、そのハッシュ値h(M)を計算する

[署名生成用ＩＣカードＢ 8060による処理]

ｽﾃｯﾌﾟ 10005B：Ａ 8051と同様に、ハッシュ値h(M)を計算する

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10006A：3005Aと同様

[署名生成用ＩＣカードＢ 8060による処理]

ステップ10006B：3005Bと同様

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10007A：3006Aと同様に(r, s_A)を計算

[署名生成用ＩＣカードＢ 8060による処理]

ステップ10007B：3006Bと同様に(r, s_B)を計算

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10008A：3007と同様に署名$s_j := (r, s)$を計算

[署名生成用ＩＣカードＢ 8060による処理]

ステップ10008B：3007と同様に署名$s_j := (r, s)$を計算

[署名生成用ＩＣカードＡ 8051による処理]

ステップ10009A：インデックスjを１増やし、M(=（j, j_S, I_j, H_j, L_j)），S_jと共に履歴保存領域に保存

[署名生成用ＩＣカードＢ 8060による処理]

ステップ10009B：インデックスjを１増やし、M(=（j, j

_S, I_j, H_j, L_j))，S_jと共に履歴保存領域に保存
[署名生成用ＩＣカードＡ 8051による処理]
ステップ10010A：S_jと共にメッセージMを署名者のコンピュータ 8031に送信
[署名生成用ＩＣカードＢ 8060による処理]
ステップ10010B：S_jと共にメッセージMを署名者のコンピュータ 8031に送信
[署名者のコンピュータ 8031による処理]
ステップ10011：メッセージMと署名S_jを購入者I_jに送信
ステップ10012：おわり
上記各ステップ中で、ＩＣカードへの入出力に際しては、必要に応じて、署名者のコンピュータまたは他のコンピュータが仲介して行うものとしてもよい。

【００９９】このようにして生成された署名S_jの検証は、たとえば、次のように行う。なお、以下の処理は購入者側において行われてもよいし、購入者あるいはその他の者から依頼を受けた調停者によって行われてもよい。以下ではこれらを総称して検証者と呼ぶ。

【０１００】検証者はまず、利用者のPC 1011と同様に構成された検証用コンピュータを用いて署名者の公開鍵 8019を使って、公知のディジタル署名検証技術により、署名S_jの正当性を検証する。正当性が確認されなければ署名S_jが正当な署名であるとは認められない。正当性が確認された場合、必要に応じてさらに詳細に検証を行うため、次の手順に進む。次の手順を実行すべき必要性としては、たとえば、署名S_jの生成時に利用された署名者の秘密鍵情報が漏えいするなどの原因により、公知のディジタル署名技術の信頼性が失われ、前記の手順だけでは十分でない場合や、メッセージMの価値が特に高く入念に検証する必要がある場合などが挙げられる。

【０１０１】次に、検証者は、署名者から署名履歴の提出を受け、その署名履歴の中に、署名S_jが含まれていることを確認する。もし含まれていなければ署名S_jが正当な署名であるとは認められない。

【０１０２】署名S_jが含まれていた場合、署名S_jの生成に係わった２枚のICカード(仮にICカードAとICカードBであったとする)のどちらか一方がS_jの直後に生成した署名S_j'に関する署名履歴を探す(これはインデックスを利用すれば特定可能である)。この署名履歴に含まれるLink dataの前半分または後半分が、署名S_jに関する署名履歴に含まれるデータから計算された$H(I_j, H_j, S_j, L_j)$に一致しているかどうかを確認する。もし一致していなければ、もう一方のICカードがS_jの直後に生成した署名履歴を探し、同様に調べ、一致していなければ署名S_jが正当な署名であるとは認められない。

【０１０３】一致した場合、署名S_j'に対して同様に確認処理を繰り返し、何らかの理由によりあらかじめ信頼できるということが判明している署名(たとえば、既にマスメディア等を通じて公表されている署名や、信頼できる第３者機関によって保証されているデータ、あるいは、署名S_jの検証に関して、利害関係のない第３者によって内容が保証されている署名など)に至るまでの連鎖が確認できれば署名S_jは正当なものと認められる。そうでなければ正当な署名であるとは認められない。

【０１０４】以上に示されるように、本検証方法に従えば、文献１２に示された、現在のディジタル署名システムの安全性が何らかの理由により脆弱化した場合にも安全性を確保できる。さらに、署名履歴の一部が失われたり、利用不可能になったりした場合にも、その正当性が確認できるという利点が得られる。これは、特に長期にわたり履歴を管理する必要があるような場合にもより頑健な方法およびそれを用いたシステムを実現できるという効果がある。

【０１０５】（第５の実施例）本発明の特徴の一つである、一つ前に生成した署名履歴だけでなくそれ以前の署名履歴も反映して署名を生成する、という方法は、一つの装置を用いた署名生成システム、タイムスタンプシステムを、履歴紛失などに対して、より頑健にする方法として応用することができる。例えば、文献１２に記された署名システムで、署名生成時に、一つ前に生成した署名履歴の他、ｎ回前までのｎ個の署名履歴を反映させるようにし、検証時にはそれらn個のうちの少なくとも一つの連鎖が確認できれば正しいと認めるようにすると、履歴を長期にわたって管理している間に、連続したn個の署名履歴が失われない限り、履歴紛失などに対して、より頑健なシステムを構成することができる。ｎ個の署名履歴を反映させるように署名生成を行うためには、上記第４実施例中のステップ10004A、10005Aに示したのと同様に署名対象データを計算すればよい。

【０１０６】検証時には、ｎ個すべての連鎖が確認できたときのみ正しいと認めるようにすれば、履歴紛失への耐性は得られないが、署名の改ざんがより困難になるという点では優れている。なぜなら、ｎ個すべての連鎖を正しく保つためには、与えられたすなわち、検証対象となる署名に含まれているｎ個のハッシュ値それぞれに対する原像(pre-image)を求める必要があるからである。

【０１０７】また、ｎ個中のｋ個(ｎ＞ｋ)の連鎖が確認できればその連鎖は正しいと認めるようにしてもよい。このようにすれば、署名改ざんの困難性を向上しつつ、履歴紛失への耐性を向上させることができる。

【０１０８】これら検証のためのルールは、署名生成時、あるいは、システム構築時にあらかじめ決めておいてもよいし、また、実際に署名検証する必要が生じたときに、その時点における、技術環境、システム全体に求められるセキュリティの程度、などに応じて決められるようにしてもよい。

【０１０９】また、署名生成時に、複数の履歴データを反映させる方法としては、履歴データを結合(連接)する方法以外にも、上記検証のためのルールにあわせて、検証が可能であるような他の方法を用いても良い。たとえ

ば、排他的論理和（XOR）や加算を用いてもよいし、ハッシュ関数を用いても良い。

【０１１０】上記文献５～文献７は、タイムスタンプサービスにおいて、複数のタイムスタンプ間の連鎖の確認を高速化を目的として、木構造をもったLinking Protocolを構成する方法を開示しているが、本実施例によれば、上記文献５～文献７が開示していない履歴紛失への耐性向上や、署名の改ざんのより一層の困難化などが可能となる。

【０１１１】上記各実施例では各署名対象データに時刻情報を加えてディジタル署名を行うタイムスタンプサービスを例にとって説明した。しかし、本発明を、時刻情報を加えずにディジタル署名を施し、履歴からその対象データが存在したことを証明するサービスに用いることも可能である。

【０１１２】

【発明の効果】以上説明したように、本発明によれば、ディジタル署名の生成、利用に際し、安全性と利便性を確保することが可能になる。

【図面の簡単な説明】

【図１】本発明の第１実施例が適用されたシステムの概略図である。

【図２】本発明の第１実施例におけるタイムスタンプ発行処理フロー図である。

【図３】本発明の第１実施例における概略処理フロー中のステップ2006の処理の詳細フロー図である。

【図４】本発明の第１実施例における概略処理フロー中のステップ2007の処理の詳細フロー図である。

【図５】本発明の第１実施例における各署名生成用コンピュータ内の履歴保存領域の様子を示した図である。

【図６】本発明の第２実施例における概略処理フロー中のステップ2006の処理の詳細フロー図である。

【図７】本発明の第３実施例における概略処理フロー中のステップ2006の処理の詳細フロー図である。

【図８】本発明の第４実施例が適用されたシステムの概略図である。

【図９】本発明の第４実施例における署名生成概略処理フロー図である。

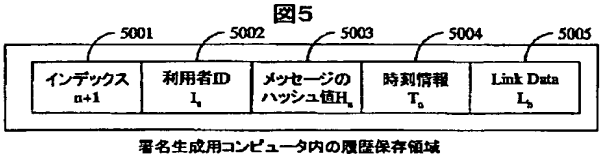
【図１０】本発明の第４実施例における署名生成概略処理フロー中のステップ9005の処理の詳細フロー図である。

【図１１】本発明の第３実施例におけるタイムスタンプ発行シーケンス時の連鎖の様子を模式的に表した図である。

【符号の説明】

1001…ネットワーク、
1002…メッセージ、
1003…タイムスタンプ付きメッセージ、
1011,1020…利用者のPC、
1012…利用者のPC 1011内のCPU、
1013…利用者のPC 1011内のメモリ、
1014…ディスプレイ、
1015…キーボード、
1016…I/O、
1017…利用者のID、
1018…プログラムPROG1、
1019…タイムスタンプ発行局の公開鍵、
1031…タイムスタンプ発行局の制御用コンピュータ、
1032…制御用コンピュータ1031内のCPU、
1033…制御用コンピュータ1031内のメモリ、
1034…ディスプレイ、
1035…キーボード、
1036…I/O(ネットワーク用）、
1037…I/O(LAN用）、
1038…プログラムPROG2、
1040…タイムスタンプ発行局側LAN、
1051,1060,1070…署名生成用コンピュータ、
1052…署名生成用コンピュータ1051内のCPU、
1053…署名生成用コンピュータ1051内のメモリ、
1054…I/O、
1055…署名生成用コンピュータのID、
1056…プログラムPROG3、
1057…秘密鍵生成情報、
1058…履歴保存領域、
1059…タイマー。

【図５】



【図１】

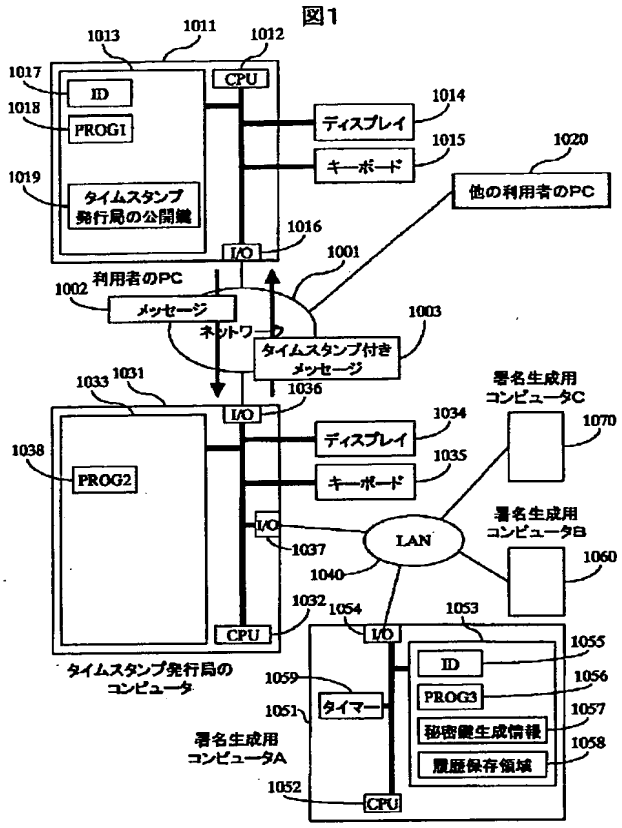


図1
1013
1011
1012
1017
ID
CPU
1018
PROG1
1014
ディスプレイ
1015
キーボード
1020
他の利用者のPC
1019
タイムスタンプ
発行局の公開鍵
1016
I/O
1001
利用者のPC
1002
メッセージ
ネットワーク
1003
タイムスタンプ付き
メッセージ
1033
1031
1036
I/O
署名生成用
コンピュータC
1034
ディスプレイ
1070
1038
PROG2
1035
キーボード
I/O
署名生成用
コンピュータB
1037
LAN
1040
1060
1032
1054
1053
CPU
I/O
1055
ID
1059
タイマー
1056
PROG3
1051
1057
秘密鍵生成情報
署名生成用
コンピュータA
1058
履歴保存領域
1052
CPU
タイムスタンプ発行局の
コンピュータ

【図２】

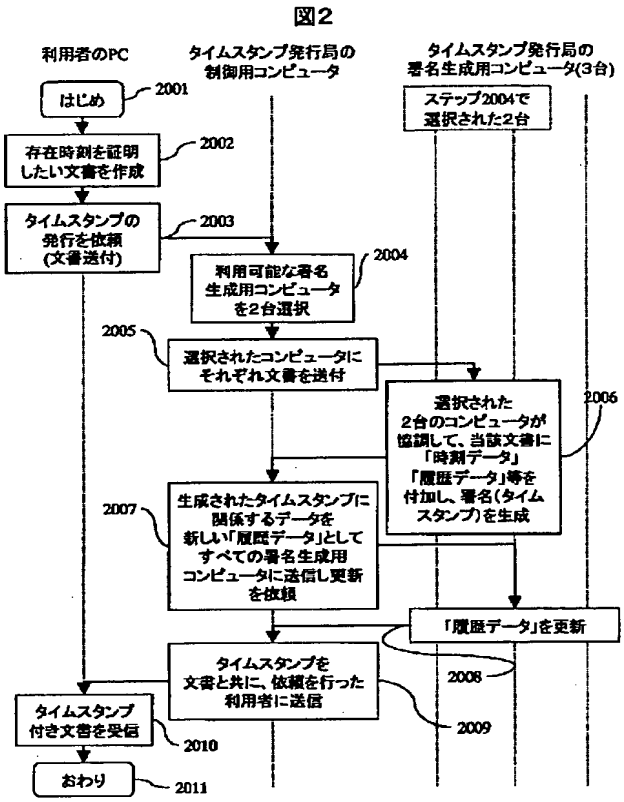


図2
利用者のPC
タイムスタンプ発行局の
制御用コンピュータ
タイムスタンプ発行局の
署名生成用コンピュータ(3台)
はじめ
2001
ステップ2004で
選択された2台
存在時刻を証明
したい文書を作成
2002
タイムスタンプの
発行を依頼
(文書送付)
2003
利用可能な署名
生成用コンピュータ
を2台選択
2004
2005
選択されたコンピュータに
それぞれ文書を送付
選択された
2台のコンピュータが
協調して、当該文書に
「時刻データ」
「履歴データ」等を
付加し、署名(タイム
スタンプ)を生成
2006
2007
生成されたタイムスタンプに
関係するデータを
新しい「履歴データ」として
すべての署名生成用
コンピュータに送信し更新
を依頼
「履歴データ」を更新
タイムスタンプを
文書と共に、依頼を行った
利用者に送信
2008
2009
タイムスタンプ
付き文書を受信
2010
おわり
2011

【図４】

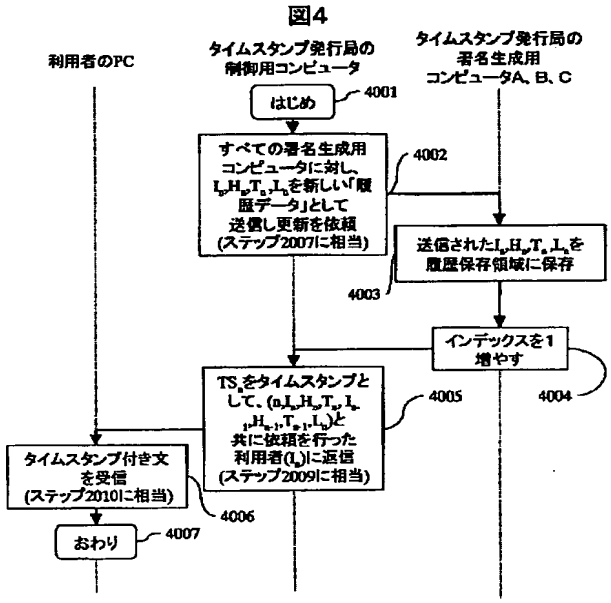
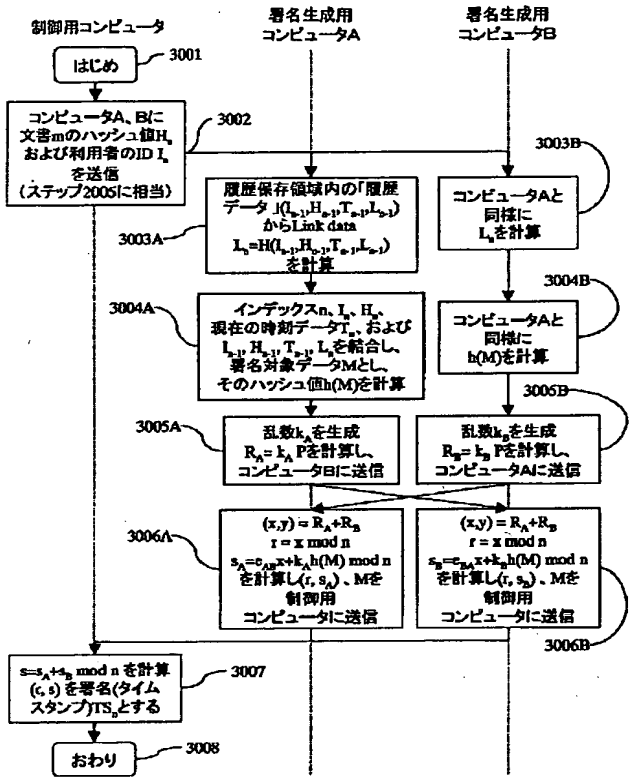


図4
利用者のPC
タイムスタンプ発行局の
制御用コンピュータ
タイムスタンプ発行局の
署名生成用
コンピュータA、B、C
はじめ
4001
すべての署名生成用
コンピュータに対し、
I_0,H_0,T_0,I_0を新しい「履
歴データ」として
送信し更新を依頼
(ステップ2007に相当)
4002
送信されたI_0,H_0,T_0,I_0を
履歴保存領域に保存
4003
インデックスを1
増やす
4004
TS_nをタイムスタンプと
して、$(n,I_n,H_n,T_n,I_{n-1},H_{n-1},T_{n-1},I_{n-1})$と
共に依頼を行った
利用者(I_n)に返信
(ステップ2009に相当)
4005
タイムスタンプ付き文
を受信
(ステップ2010に相当)
4006
おわり
4007

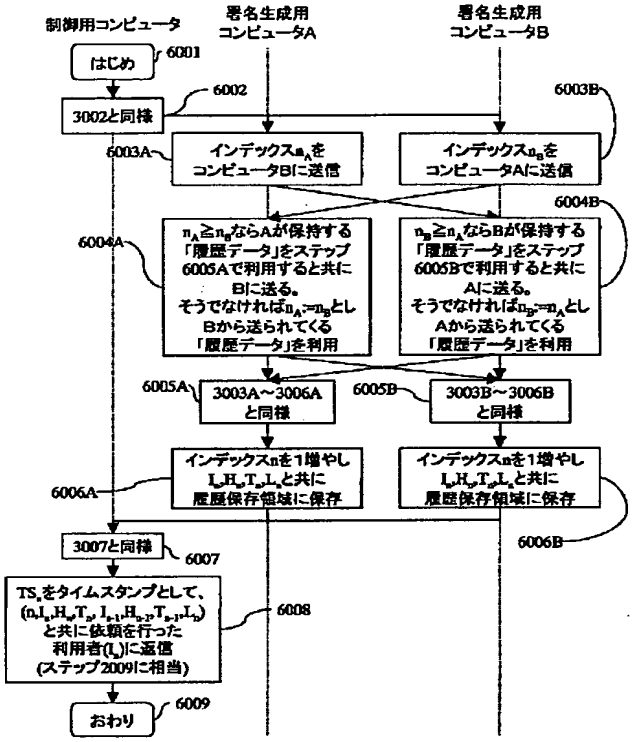
【図３】

図3



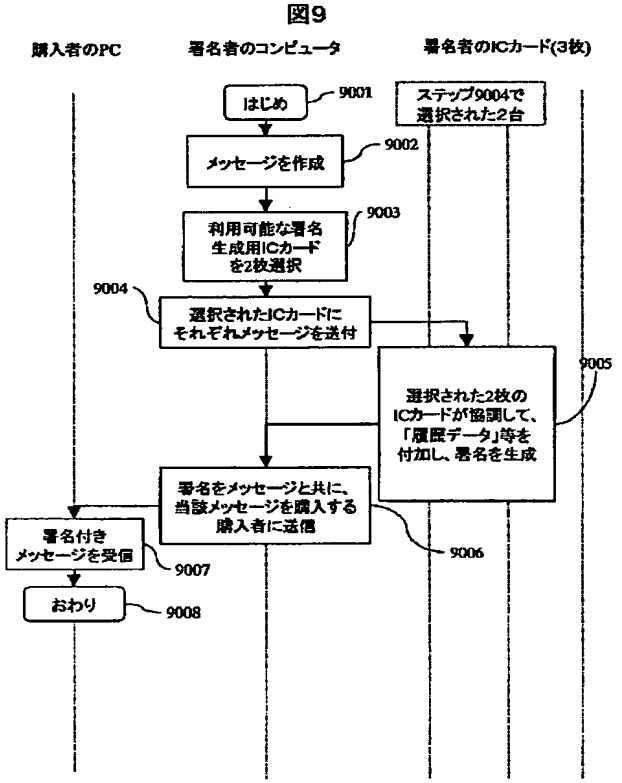
【図６】

図6



【図９】

図9



【図７】

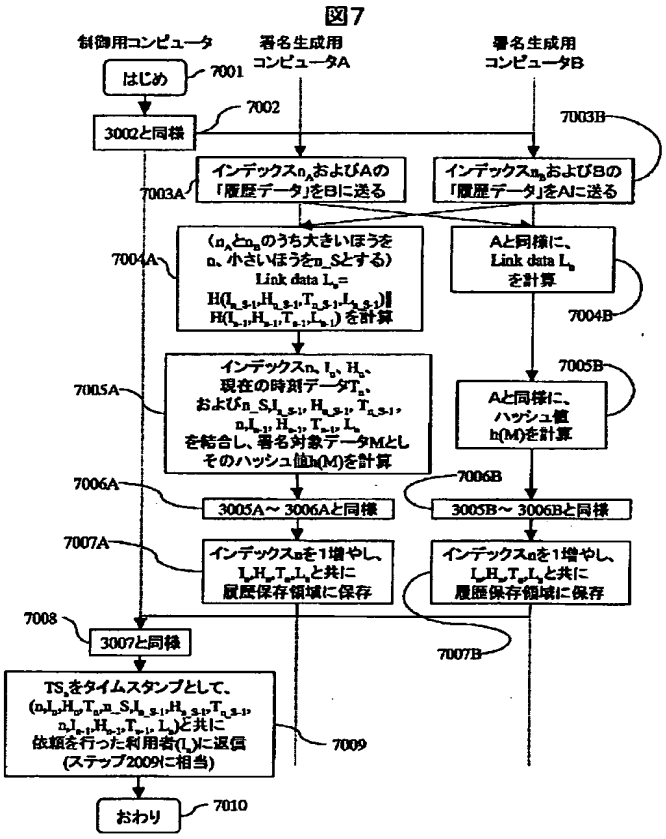


図7
制御用コンピュータ
署名生成用コンピュータA
署名生成用コンピュータB
はじめ
7001
3002と同様
7002
7003B
インデックスnAおよびAの「履歴データ」をBに送る
インデックスnBおよびBの「履歴データ」をAに送る
7003A
(nAとnBのうち大きいほうをn、小さいほうをn_Sとする)
Link data Ln=
H(In_S-1,Hn_S-1,Tn_S-1,In_S-1)‖
H(In-1,Hn-1,Tn-1,Ln-1)を計算
7004A
Aと同様に、Link data Lnを計算
7004B
インデックスn、In、Hn、現在の時刻データTn、およびn_S,In_S-1,Hn_S-1,Tn_S-1,n,In-1,Hn-1,Tn-1,Lnを結合し、署名対象データMとしそのハッシュ値h(M)を計算
7005A
Aと同様に、ハッシュ値h(M)を計算
7005B
7006A
3005A～3006Aと同様
7006B
3005B～3006Bと同様
7007A
インデックスnを1増やし、LnHnTnLnと共に履歴保存領域に保存
インデックスnを1増やし、LnHnTnLnと共に履歴保存領域に保存
7007B
7008
3007と同様
TSnをタイムスタンプとして、(n,In,Hn,Tn,n_S,In_S-1,Hn_S-1,Tn_S-1,n,In-1,Hn-1,Tn-1,Ln)と共に依頼を行った利用者(In)に返信(ステップ2009に相当)
7009
おわり
7010

【図８】

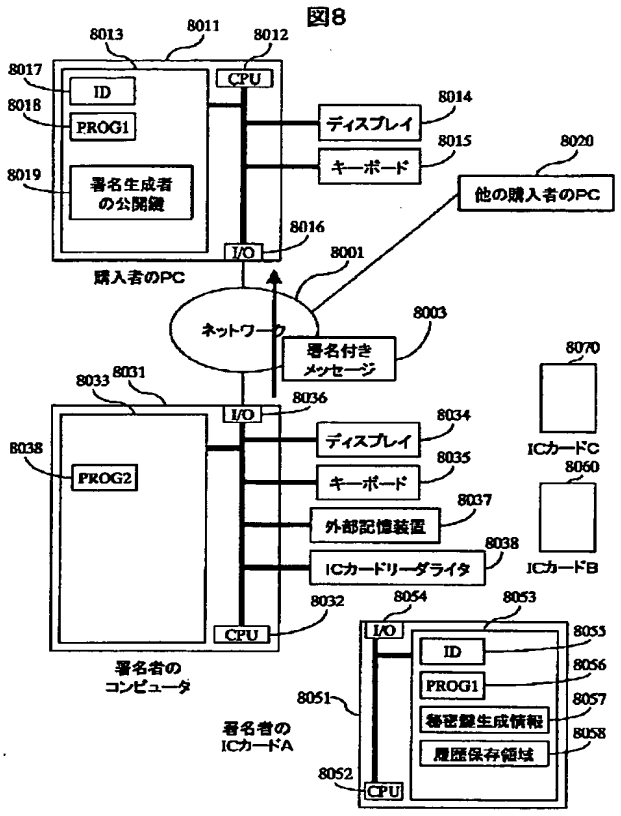


図8
8013
8011
8012
8017
ID
CPU
8018
PROG1
8014
ディスプレイ
8015
キーボード
8020
他の購入者のPC
8019
署名生成者の公開鍵
8016
I/O
購入者のPC
8001
ネットワーク
8003
署名付きメッセージ
8070
ICカードC
8033
8031
8036
8034
ディスプレイ
8038
PROG2
8035
キーボード
8060
8037
外部記憶装置
ICカードB
8038
ICカードリーダライタ
8032
8054
8053
I/O
8055
ID
8056
PROG1
8051
8057
暗号鍵生成情報
8058
履歴保存領域
署名者のICカードA
8052
CPU
署名者のコンピュータ

【図１１】

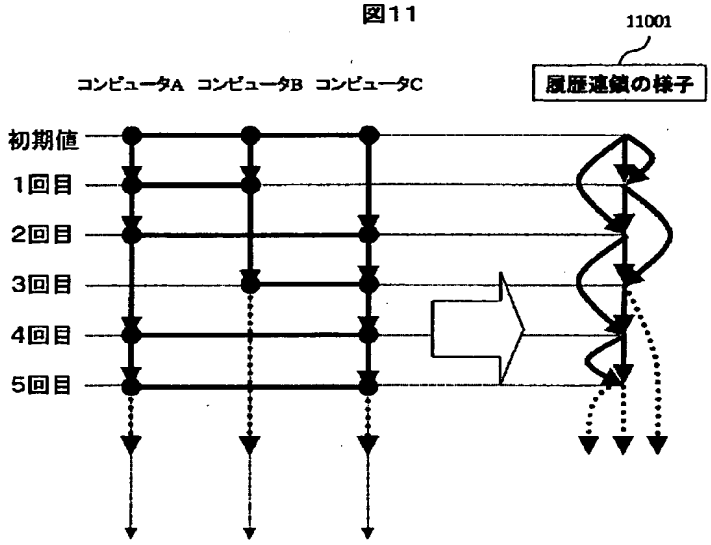
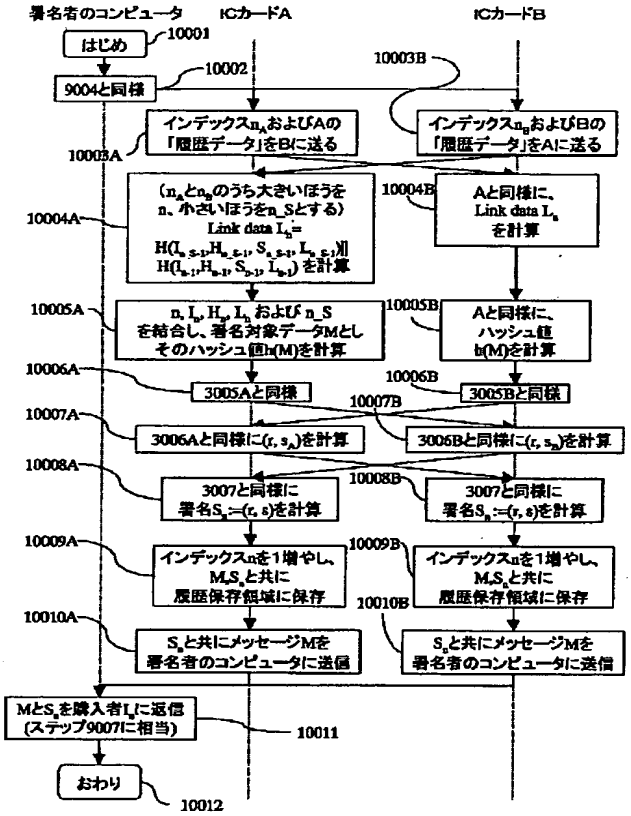


図11
11001
コンピュータA　コンピュータB　コンピュータC
履歴連鎖の様子
初期値
1回目
2回目
3回目
4回目
5回目

【図１０】

図10



フロントページの続き

(51)Int.Cl.[7]		識別記号	FI			テーマコード(参考)
Ｇ０６Ｆ	17/60	１４０	Ｇ０６Ｆ	17/60	１４０	
		５１２			５１２	
	19/00	１４０		19/00	１４０	

(72)発明者　宝木　和夫
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　洲崎　誠一
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　三島　久典
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　松木　武
神奈川県川崎市幸区鹿島田890番地　株式会社日立製作所情報サービス事業部内
(72)発明者　竹内　国人
神奈川県川崎市幸区鹿島田890番地　株式会社日立製作所金融システム事業部内
(72)発明者　岩村　充
東京都練馬区中村２－14－17
(72)発明者　松本　勉
横浜市青葉区柿の木台13－45

Ｆターム(参考)　5J104 AA09 AA11 JA25 LA03 LA06 NA02 NA05